

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日 2 0 0 2 年 1 2 月 1 6 日
Date of Application:

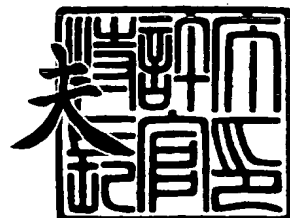
出 願 番 号 特 願 2 0 0 2 - 3 6 3 8 9 4
Application Number:
[ST. 10/C]: [J P 2 0 0 2 - 3 6 3 8 9 4]

出 願 人 株式会社エヌ・ティ・ティ・ドコモ
Applicant(s):

2 0 0 3 年 1 1 月 1 8 日

特許庁長官
Commissioner,
Japan Patent Office

今 井 康



出証番号 出証特 2 0 0 3 - 3 0 9 5 1 6 4

【書類名】 特許願

【整理番号】 002238

【提出日】 平成14年12月16日

【あて先】 特許庁長官殿

【国際特許分類】 H04M 11/00

【発明者】

【住所又は居所】 東京都千代田区永田町二丁目 1 1 番 1 号 株式会社エヌ
・ ティ ・ ティ ・ ドコモ内

【氏名】 石川 太朗

【発明者】

【住所又は居所】 東京都千代田区永田町二丁目 1 1 番 1 号 株式会社エヌ
・ ティ ・ ティ ・ ドコモ内

【氏名】 稲村 浩

【発明者】

【住所又は居所】 東京都千代田区永田町二丁目 1 1 番 1 号 株式会社エヌ
・ ティ ・ ティ ・ ドコモ内

【氏名】 三宅 基治

【発明者】

【住所又は居所】 東京都千代田区永田町二丁目 1 1 番 1 号 株式会社エヌ
・ ティ ・ ティ ・ ドコモ内

【氏名】 横田 和久

【発明者】

【住所又は居所】 東京都千代田区永田町二丁目 1 1 番 1 号 株式会社エヌ
・ ティ ・ ティ ・ ドコモ内

【氏名】 高橋 修

【特許出願人】

【識別番号】 392026693

【氏名又は名称】 株式会社エヌ ・ ティ ・ ティ ・ ドコモ

【代理人】**【識別番号】** 100066980**【弁理士】****【氏名又は名称】** 森 哲也**【選任した代理人】****【識別番号】** 100075579**【弁理士】****【氏名又は名称】** 内藤 嘉昭**【選任した代理人】****【識別番号】** 100103850**【弁理士】****【氏名又は名称】** 崔 秀▲てつ▼**【手数料の表示】****【予納台帳番号】** 001638**【納付金額】** 21,000円**【提出物件の目録】****【物件名】** 明細書 1**【物件名】** 図面 1**【物件名】** 要約書 1**【包括委任状番号】** 0016816**【プルーフの要否】** 要

【書類名】 明細書

【発明の名称】 プロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置

【特許請求の範囲】

【請求項 1】 所定通信プロトコルに従って少なくとも 1 以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出方法であって、

前記送受信端末間における通信の間に送受されるパケットを取得することにより、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出ステップと、

前記算出ステップにおいて算出された状態情報と、前記少なくとも 1 以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較ステップと、

を有し、前記比較ステップにおける比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とするプロトコル不具合自動検出方法。

【請求項 2】 前記通信プロトコルに従って前記送受信端末において送受されるパケットに基づいて行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定ステップを、更に、含み、

前記不具合情報は、前記不具合がある場合における、前記算出ステップにおいて算出される状態情報と前記正常情報との関係を規定することを特徴とする請求項 1 に記載のプロトコル不具合自動検出方法。

【請求項 3】 前記不具合情報は、前記状態情報と、前記送受信制御処理の不具合についてあらかじめ確認されている固定値と、の関係を規定することを特徴とする請求項 1 又は 2 に記載のプロトコル不具合自動検出方法。

【請求項 4】 前記算出ステップにおいては、前記パケットの取得のたびに、前記状態情報を更新し、

前記比較ステップにおいては、前記算出ステップにおいて更新される最新の状

態情報と、前記不具合情報と、を比較することを特徴とする請求項 1～3 のいずれか 1 項に記載のプロトコル不具合自動検出方法。

【請求項 5】 前記状態情報は、送受信パケット数の合計値、パケットサイズの最小値又は最大値、又は、送信されたパケットに対する応答パケットを受信するまでのラウンドトリップタイム等の情報であることを特徴とする請求項 1～4 のいずれか 1 項に記載のプロトコル不具合自動検出方法。

【請求項 6】 所定通信プロトコルに従って少なくとも 1 以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出装置であって、

前記送受信端末間における通信の間に送受されるパケットを取得するパケット取得手段と、

前記パケット取得手段により取得したパケットに基づいて、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出手段と、

前記算出手段により算出された状態情報と、あらかじめ蓄積される、前記少なくとも 1 以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較手段と、

を有し、前記比較手段による比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とするプロトコル不具合自動検出装置。

【請求項 7】 前記パケット取得手段により取得されるパケットに基づいて、前記通信プロトコルに従って、前記送受信端末において該取得したパケットに対して行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定手段を、更に、含み、

前記不具合情報は、前記不具合がある場合における、前記算出手段により算出される状態情報と前記正常情報との関係を規定することを特徴とする請求項 6 に記載のプロトコル不具合自動検出装置。

【請求項 8】 前記パケット取得手段により取得したパケットのヘッダ情報に基づいて、必要なパケットのみを選択して前記算出手段に転送するためのパケ

ットフィルタ手段を、更に、有することを特徴とする請求項6又は7に記載のプロトコル不具合自動検出装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明はプロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置に関し、特に新規の通信装置の導入時の不具合検出等に用いることができるプロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置に関する。

【0002】

【従来の技術】

近年インターネットアクセスが急速に増大しつつあり、TCP/IP (Transmission Control Protocol/Internet Protocol) プロトコルを実装する様々なコンピュータ、通信デバイスが開発されている。またTCP/IPプロトコルを用いる新たなアプリケーションが開発され、アプリケーションの種類が増大している。TCP/IPプロトコルを実装するコンピュータ、通信デバイスの種類が増えることでTCP/IPプロトコル実装の不具合の種類が増える可能性がある。また、従来アプリケーションでは問題のなかったTCP/IP実装において、新たなアプリケーションに利用することにより、潜在していた不具合が誘発される可能性がある。すなわち、ここでいうTCP/IPプロトコル実装の不具合としては、通信デバイス等がTCP/IPプロトコルの仕様に従った動作を行わない場合やアプリケーションにおける利用上の不備等の実装上の不備に起因してTCP/IPプロトコルにおいて予定する通信処理が行なわれない不具合が生じる場合のみでなく、新たなアプリケーション等への利用により従来の利用においては想定し得なかったTCP/IPプロトコル自体の不備・欠陥により予定する通信処理が機能しない場合があげられる。

【0003】

このようなTCP/IPプロトコル実装の不具合を判断するための従来の手法では、例えば、tcpdump (非特許文献1参照) に代表されるプロトコルア

ナライザを用いる。図6にプロトコルアナライザ8の機能ブロック図を示す。プロトコルアナライザ8は、主にネットワークに流れるパケットを収集する機能を有するものである（ネットワークインターフェース8a、パケット受信部8b）。パケット中のプロトコルヘッダを各情報区切り毎に、認識可能なテキストデータ等に翻訳し（パケット翻訳部8d）、画面出力させる（画面出力部8e）ことにより、パケットの内容を把握してプロトコルの不具合判断が可能になる。

【0004】

さらに、TCPに限ると、`tcptrace`（非特許文献2参照）に代表される解析ツールが提案され、`tcpdump`等の標準的なプロトコルアナライザにより収集したパケットの保存データから、転送データ量、再送データ量、スループット、ラウンドトリップタイムなどの統計情報を得ることができる。これらの統計情報を、画面出力等することによりプロトコルの不具合の判断材料として用いることができる。図7に解析ツール9の機能ブロック図を示す。

【0005】

【非特許文献1】

RFC2398 Some Testing Tools for TCP Implementors、[online]、[平成14年12月11日検索]、インターネット<URL:<http://www.tcpdump.org/>>

【非特許文献2】

RFC2398 Some Testing Tools for TCP Implementors、[online]、[平成14年12月11日検索]、インターネット<URL:<http://www.tcptrace.org/>>

【0006】

【発明が解決しようとする課題】

しかしながら、上述のプロトコルアナライザから得た翻訳出力や解析ツールで得た統計情報を用いたプロトコル不具合検出には不十分である。

すなわち、プロトコルアナライザは、同時に収集される複数のコネクションに

において送受されるパケットそれぞれのヘッダの翻訳のみを行う。このため、プロトコル不具合検出時には、翻訳情報から、それぞれのパケットをコネクションごとに対応付け、更にどのパケットがプロトコル固有のシーケンスのどの部分に相当するものなのかの対応付けを行うなど、煩雑な作業を行わなければならない。

【0007】

解析ツールについても、プロトコルアナライザの保存データを加工し、コネクション毎に伝送データ量、再送データ量、スループット等の統計値や、シーケンス図等のグラフを提供するが、ある程度の異常の発生については確認することができても、どのような処理の不具合によって異常が発生しているのかについての判断までも与えるものではない。このため、解析ツールが示す結果によってどのような処理の不具合であるのか等の原因特定を行うためには、解析ツールによって事後的に示される結果に基づいて、不具合が発生したと思われる時間周辺のパケットを特定し、プロトコル固有のシーケンスに沿ってパケットの構成に異常がないかを調べるなどして処理の異常を特定する必要がある。更に、通信状態等に応じてプロトコルに従った処理の内容が変化することなども考慮する必要がある、原因を特定するには、専門的な知識と煩雑な作業を要することとなる。

本発明の目的は、上述の課題に鑑みてなされたものであり、専門的な知識と煩雑な作業なしにプロトコルの不具合を検出可能なプロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置を提供することにある。

【0008】

【課題を解決するための手段】

本発明の請求項1によるプロトコル不具合自動検出方法は、所定通信プロトコルに従って少なくとも1以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出方法であって、

前記送受信端末間における通信の間に送受されるパケットを取得することにより、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出ステップと、

前記算出ステップにおいて算出された状態情報と、前記少なくとも1以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較ス

テップと、

を有し、前記比較ステップにおける比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とする。

【0009】

本発明の請求項2によるプロトコル不具合自動検出方法は、請求項1において、前記通信プロトコルに従って前記送受信端末において送受されるパケットに基づいて行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定ステップを、更に、含み、

前記不具合情報は、前記不具合がある場合における、前記算出ステップにおいて算出される状態情報と前記正常情報との関係を規定することを特徴とする。

【0010】

本発明の請求項3によるプロトコル不具合自動検出方法は、請求項1又は2において、前記不具合情報は、前記状態情報と、前記送受信制御処理の不具合についてあらかじめ確認されている固定値と、の関係を規定することを特徴とする。

本発明の請求項4によるプロトコル不具合自動検出方法は、請求項1～3のいずれか1項において、前記算出ステップにおいては、前記パケットの取得のたびに、前記状態情報を更新し、

前記比較ステップにおいては、前記算出ステップにおいて更新される最新の状態情報と、前記不具合情報と、を比較することを特徴とする。

【0011】

本発明の請求項5によるプロトコル不具合自動検出方法は、請求項1～4のいずれか1項において、前記状態情報は、送受信パケット数の合計値、パケットサイズの最小値又は最大値、又は、送信されたパケットに対する応答パケットを受信するまでのラウンドトリップタイム等の情報であることを特徴とする。

本発明の請求項6によるプロトコル不具合自動検出装置は、所定通信プロトコルに従って少なくとも1以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出装置であって、

前記送受信端末間における通信の間に送受されるパケットを取得するパケット

取得手段と、

前記パケット取得手段により取得したパケットに基づいて、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出手段と、

前記算出手段により算出された状態情報と、あらかじめ蓄積される、前記少なくとも 1 以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較手段と、

を有し、前記比較手段による比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とする。

【0012】

本発明の請求項 7 によるプロトコル不具合自動検出装置は、請求項 6 において、前記パケット取得手段により取得されるパケットに基づいて、前記通信プロトコルに従って、前記送受信端末において該取得したパケットに対して行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定手段を、更に、含み、

前記不具合情報は、前記不具合がある場合における、前記算出手段により算出される状態情報と前記正常情報との関係を規定することを特徴とする。

【0013】

本発明の請求項 8 によるプロトコル不具合自動検出装置は、請求項 6 又は 7 において、前記パケット取得手段により取得したパケットのヘッダ情報に基づいて、必要なパケットのみを選択して前記算出手段に転送するためのパケットフィルタ手段を、更に、有することを特徴とする。

【0014】

【発明の実施の形態】

次に、図面を参照して本発明の実施の形態について説明する。なお、以下の説明において参照する各図においては、他の図と同等の部分が同一符号によって示されている。

(プロトコル不具合自動検出装置の構成)

図 1 には、本実施の形態におけるプロトコル不具合自動検出装置の構成を説明

するブロック図が示されている。

【0015】

ネットワークインターフェイス 1 a は、外部ネットワークと通信を行う機能を有する。

パケット受信部 1 b は、ネットワークインターフェイス 1 a に到着したパケットを受信し、保存が必要な場合はデータ保存のためにパケット保存・読込部 1 k にパケットを転送し、保存が必要でない場合は、パケットフィルタ・解析部 1 c にする機能を有する。

【0016】

パケット保存・読込部 1 k は、パケットの保存が必要な場合、パケット受信部 1 b で受信したパケットの保存を行い、保存済みのパケットデータの不具合解析を行う場合には、保存済みのパケットデータを、パケット受信部 1 b を介してパケットフィルタ・解析部 1 c に転送する機能を有する。

パケットフィルタ・解析部 1 c は、パケット受信部 1 b から受け取ったパケットのヘッダ情報を解析し、必要な種類のパケット以外を廃棄する機能と、必要な種類のパケットのヘッダ情報及びペイロード情報を不具合比較判定部及びコネクション情報算出部 1 d に転送する機能を有する。

【0017】

コネクション情報算出部 1 d は、パケットフィルタ・解析部 1 c を介してパケットのヘッダ情報及びペイロード情報を受け取り、TCPコネクション情報を作成し、コネクション情報保存部 1 e に保存する機能を有する。

TCPコネクション情報は、本実施の形態においては、TCPコネクションが設定されている間に送受されるパケットを取得することにより、TCP/IPプロトコルに従った送受信制御の結果に対応すべきパケットの送受信状態に関する状態情報である。本実施の形態においては、パケットのヘッダ情報、ペイロード情報やパケットの送受信事象の発生といったパケットそのものを解析することにより取得し、当該コネクションに該当するパケットのヘッダ情報及びペイロード情報を受け取る度に更新される。TCPコネクション情報に含まれる情報としては、送信パケット数、再送パケット数、SACKブロック数の各種合計値、最小

パケットサイズ、再送間隔の最大値等のスループット、ラウンドトリップタイム等の各種評価値等がある。

【0018】

コネクション情報保存部 1 e は、コネクション情報算出部 1 d にて作成されたため情報を保存する機能を有する。

正常情報推定部 1 f は、コネクション情報算出部 1 d を介してパケットのヘッダ情報及びペイロード情報を受け取り、当該パケットのヘッダ情報等に基づいて、当該パケットの送信元又は送信先の送受信端末において行われるべき送受信制御処理を特定し、当該特定された送受信制御処理が正常に行われた場合におけるその処理結果に対応すべき正常情報を推定する。正常情報としては、例えば、TCP コネクションを確立し、TCP/IP プロトコルに従った送受信制御を行う送受信端末において、当該制御を実施する TCP において用いられる `cwnd`、`ssthresh`、`srtt`、`rttvar` 等の内部変数や TCP の状態遷移ダイアグラムの状態推定値などがある。これらの推定された正常情報は、正常情報保存部 1 g に保存される。

【0019】

正常情報保存部 1 g は、正常情報推定部 1 f において推定された正常情報を保存する機能を有する。

不具合比較判定部 1 h は、パケットフィルタ・解析部 1 c の解析結果と、正常情報保存部 1 g に保存される正常情報と、不具合情報保存部 1 i に保存される不具合情報と、コネクション情報保存部 1 e に保存される TCP コネクション情報と、を比較することにより不具合の発生している処理を検出する。この比較判定の結果は、判定結果出力部 1 j に転送される。

【0020】

不具合情報保存部 1 i は、これまで知られているプロトコルにおける少なくとも 1 以上の処理のそれぞれの不具合を特徴付ける情報を保存する機能を有する。不具合を特徴付けるデータの具体例としては、TCP コネクション情報に関する条件式、パケットヘッダ情報に関する条件式、又は、それらの組み合わせが挙げられる。TCP コネクション情報に関する条件式としては、例えば、後述するよ

うに、TCPコネクション情報に保持される値と推定された正常情報の値、あるいは、TCPコネクション情報が異常時にとる固定値・算出値、との大小関係を規定する式などが挙げられる。また、パケットヘッダ情報に関する条件式は、例えば不正なパケットの構成を規定するなど、パケットそのものの構成の不具合に関する条件式である。

【0021】

判定結果出力部1jは、不具合比較判定部1hで行った判定結果を出力する機能を有する。

(プロトコル不具合自動検出装置及びプロトコル不具合自動検出方法の実施例)

以下、図1のプロトコル不具合自動検出装置を用いた不具合検出の一例として、TCPによるパケットの送受信制御の1つである輻輳制御のためのFast Retransmit/Fast Recovery アルゴリズム(RFC 2581 TCP Congestion Control)を行わない不具合を検出する方法を説明する。

【0022】

図2には、本実施の形態にかかる上述のプロトコル不具合自動検出装置が用いられるネットワークの全体構成を説明する図が示されている。

サーバ21は、ルータ23、インターネットI、ルータ24を介してクライアント22と通信している。サーバ21はFast Retransmit/Fast Recoveryアルゴリズム正しく行わない不具合がある。

【0023】

プロトコル不具合自動検出装置1は、サーバ21と同じイーサネット(登録商標)セグメントに接続されているため、サーバ21が送受信するパケットをすべて受信することができる。

コネクション情報算出部1dは、統計値snd__uma、snd__maxを扱う。snd__umaは、これまでにサーバ21に応答確認されたデータセグメントの値、snd__maxは、サーバ21から送信されたデータセグメントのシーケンス番号の最大値を示す統計値である。ゆえに、(snd__max - snd__uma)の統計値は、正常時には、サーバ21における輻輳制御処理の結果

に対応した値をとる。

【0024】

正常情報推定部 1 f は、サーバ 2 1 に送信されたパケットを受信する度に、Fast retransmit/Fast Recovery アルゴリズムに基づいて、cwnd、sssthresh を推定し、それぞれ正常情報保存部 1 g の snd_cwnd、snd_ssllhresh に保存する。

正常情報保存部 1 g は、正常情報として snd_cwnd、snd_ssllhresh を扱う。

【0025】

不具合情報保存部 1 i は Fast Retransmit/Fast Recovery アルゴリズムを行わない不具合を特徴づける条件式 (1) を保有する。

$$(snd_max - snd_uma) > snd_cwnd \cdots (1)$$

不具合比較判定部 1 h は、不具合情報保存部 1 i に保存されている条件式 (1) が満たされると、正しく Fast retransmit/Fast Recovery アルゴリズムが行われない不具合を観測した旨表示する。

【0026】

正常情報保存部 1 g の snd_cwnd が 43800、snd_ssllhresh が 65535 で、これらの推測値が正しく推測された状態において、図 3 に示すパケットを受信すると、snd_max、snd_uma、(snd_max - snd_uma)、及び、snd_cwnd は、それぞれ図 4 のように変化する。

【0027】

Fast Retransmit/Fast Recovery アルゴリズムを実装する正常なサーバ 2 1 の TCP では、新たな ACK を受信するたびに輻輳ウィンドウサイズを規定する内部変数 cwnd の値を増やし、3 つの重複 ACK 受信すると (輻輳によるパケットロスが確認されると)、cwnd をこれまでの値の半分にし、さらに 3 セグメント分増やす。2 つまでの重複 ACK に対しては cwnd の更新は行わない。このような制御により、輻輳時におけるパケット転

送量の制御を行っている。

【0028】

図4に示される、時刻15:37:21.667007のACK受信P1、及び、時刻15:37:21.697003のACK受信P2は、重複ACK受信であり、正常なTCPでは、これらのACK受信に対して、`cwnd`の更新は行わない。

時刻15:37:21.727007のACK受信P3は3つ目の重複ACK受信でありこのACK受信P3に対して、`cwnd`をこれまでのこの半分にし、さらに3セグメント分増やす操作を行う。これ以後の重複ACKに対しては、重複ACKを受け取るたびに1セグメント分増やす。

【0029】

この正常なアルゴリズムに従った`cwnd`の推測値`snd__cwnd`は、ACK受信P1及びACK受信P2によっては変化せず、ACK受信P3によって、これまでの値の半分+3セグメント分 ($46720/2 + 1460 \times 3 = 27740$) に更新され、以後重複ACK受信P4、P5、P6、、、のたびに、29200、30660、32120、、、と1セグメント分増やす。

【0030】

これに対し、(`snd__max` - `snd__uma`) は2つまでの重複ACK受信P1、P2に連動して増加し(値45260、46720)、3つ目の重複ACK受信P3に対しても値が小さくならず(値46720)、以後の重複ACK受信P4、P5、P6、、、に連動して値が増加(値48180、49640、51100、、、)している。

【0031】

そのため、15:37:21.727007のACK受信P3以降、条件式(1)が成立し、判定結果出力部1jに正しく`Fast retransmit/Fast Recovery`アルゴリズムが行わない不具合を観測した旨が表示される。

この例において実現されているプロトコル不具合自動検出方法のフローチャートが図5に示されている。以下、図2をも参照しながら説明する。

【0032】

同図のステップS101においては、サーバ21及びクライアント22間で送受されるパケットがプロトコル不具合自動検出装置1において取得されることにより、TCPに従った送受信制御の結果に対応すべきパケットの送受信状態に関する状態情報、本例においては、`snd__max`及び`snd__uma`が算出される。

【0033】

同図のステップS102においては、TCPに従ってサーバ21において送受されるパケットに基づいて行われるべき輻輳制御処理が正常に行われた処理結果に対応すべき`snd__cwnd`がプロトコル不具合自動検出装置1において推定される。

同図のステップS103においては、ステップS101において算出された状態情報（`snd__max`、`snd__uma`）と、輻輳制御処理の不具合を特徴付ける不具合情報との比較により不具合の検出が行われる。ここで不具合情報は、ステップS101で算出された状態情報と、ステップS102で推定された正常情報との関係が規定されるので、上記条件式（1）を満たすか否かの比較が行われる。

【0034】

上記ステップS101～ステップS103の処理が、サーバ21及びクライアント22間における通信の間にわたって、繰り返される。

（プロトコルの不具合の検出の具体例）

また、上述の実施例のほかにも、プロトコルの不具合の検出として以下のような例が挙げられる。

【0035】

例えば、コネクションで送受されるパケットの取得時刻の算出により、次のような不具合を検出することができる。例として、クライアントのTCPからサーバへのHTTP（HyperText Transfer Protocol）接続処理においては、TCPコネクションの確立後2秒経てから、HTTP GET要求パケットが送信されるという不具合がある。通常においては、HTTP

GETは、TCPコネクションの確立直後に送信される。

【0036】

この場合には、例えば、以下のように不具合を検出する。

まず、例えば上述のプロトコル不具合自動検出装置のコネクション情報算出部において、active openによるコネクション確立を検出し、そのコネクション確立時刻を記録する。すなわち、クライアントからサーバへSYNパケット、これに応答するサーバからクライアントへのACK+SYNパケット、再びクライアントからサーバへのACKパケットの3つのパケットがやり取りされるのを検出する。このパケット取得のたびに取得・更新の必要な情報については、コネクション情報算出部にあらかじめ登録しておく。

【0037】

次に、クライアントからサーバへの最初のパケットの送信を検出し、その最初のデータ送信時刻を記録する。

次に、上述の不具合比較判定部において、例えば、以下の条件式(2)を規定する不具合情報に基づいて、比較判定を行うことにより、HTTP接続処理の不具合を検出することができる。この比較判定は、例えば、最初のデータ送信時刻等の取得・更新を契機として、あるいは、パケット取得の度に行う。

(最初のデータ送信時刻－コネクション確立時刻) > 2秒 … (2)

また、次のような不具合を検出することもできる。例えば、TCPの再送処理について、タイムアウトによる再送の初期値が60秒となる不具合がある。通常においては、再送処理を行なう通信装置においてラウンドトリップタイムを計測しており、計測したラウンドトリップタイムを基に、逐次タイムアウトの値を計算するため、その結果としてタイムアウトの時刻が60秒に設定されることはほとんどない。

【0038】

この場合には、例えば、以下のように再送処理の不具合を検出する。

コネクション情報算出部において、送信データセグメントを取得すると、その送信時刻を記録する。

次に、コネクション情報算出部において、タイムアウトにより再送された送信

パケットを取得したら、その再送時刻を記録する。

【0039】

次に、不具合比較判定部において、不具合情報に基づいて、送信時刻から再送時刻を引き算することにより算出された再送時間が60秒である場合には不具合と判定する。

更に、例えば、コネクションで送受されるパケットの構成に関する不具合情報により、次のような不具合を検出することができる。例として、TCPヘッダのオプションフィールドのパディングが必要以上に存在するという不具合（オプションフィールドに設定されるべき情報が設定されていないなど）がある。通常は、オプションデータが設定されるため、TCPオプションフィールドのパディングは、3バイト以下である。

【0040】

この場合には、例えば、以下のように不具合を検出する。

まず、コネクション情報算出部又はパケットフィルタ・解析部にて、TCPヘッダのオプションフィールドを解釈し、オプションデータが占めるバイト数を求める。

次に、不具合比較判定部において、オプションフィールドの総バイト数からオプションデータが占めるバイト数を減算することにより算出される使用されない領域のバイト数が4バイト以上である場合には、不具合と判定し、警告を表示する。

【0041】

【発明の効果】

以上説明詳細に説明したように、本発明にかかる請求項1に記載のプロトコル不具合自動検出方法、及び、本発明にかかる請求項6に記載のプロトコル不具合自動検出装置によれば、あらかじめ定められたプロトコルにしたがって行われる少なくとも1以上の処理の不具合を特徴付けるデータと、当該不具合データに対応する実際の通信状態と、の比較により、煩雑な処理を行なうことなく不具合が発生している処理を特定することができる。

【0042】

本発明にかかる請求項 2 に記載のプロトコル不具合自動検出方法、及び、本発明にかかる請求項 7 に記載のプロトコル不具合自動検出装置によれば、プロトコルによりあらかじめ定められる正常処理の推定値と、当該推定値に対応する実際のパケットの送受信状態と、を比較するため、より正確に、かつ、汎用的にプロトコルの不具合を検出することができる。また、入力には実際のパケットを用いるので、通信状態に応じて制御内容が変化するような処理の不具合についても検出することができる。

【図面の簡単な説明】

【図 1】

本実施の形態におけるプロトコル不具合自動検出装置の構成を説明するブロック図である。

【図 2】

本実施の形態にかかるプロトコル不具合自動検出装置が用いられるネットワークの全体構成を説明する図である。

【図 3】

プロトコル不具合自動検出装置において取得するパケットを説明する図である。

【図 4】

プロトコル自動検出装置におけるパケットの取得に基づいて、プロトコル不具合自動検出装置で算出あるいは推定される情報を説明する図である。

【図 5】

本実施の形態にかかるプロトコル不具合自動検出方法を説明するフローチャートである。

【図 6】

従来のプロトコルアナライザの機能ブロック図である。

【図 7】

従来の解析ツールの機能ブロック図である。

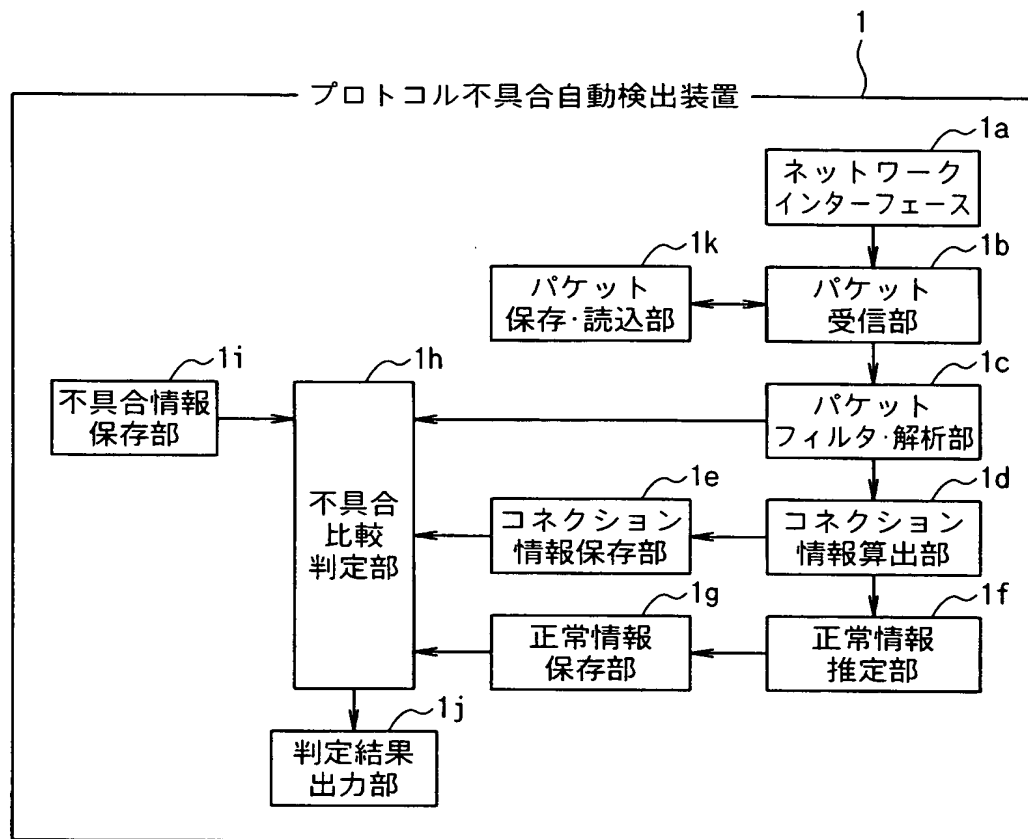
【符号の説明】

1 プロトコル不具合自動検出装置

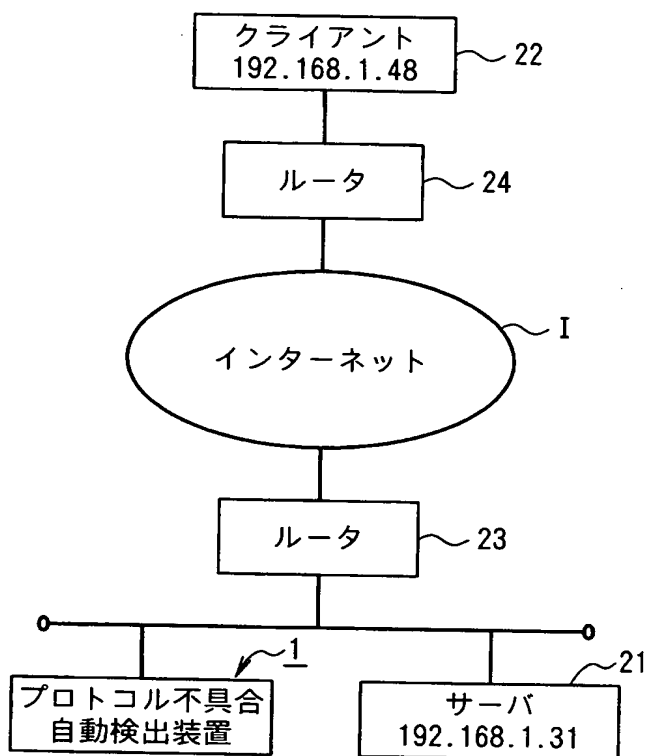
1 a	ネットワークインターフェイス
1 b	パケット受信部
1 c	解析部
1 d	コネクション情報算出部
1 e	コネクション情報保存部
1 f	正常情報推定部
1 g	正常情報保存部
1 k	読込部
1 j	判定結果出力部
1 i	不具合情報保存部
1 h	不具合比較判定部
8	プロトコルアナライザ
8 a	ネットワークインターフェース
8 b	パケット受信部
8 d	パケット翻訳部
8 e	画面出力部
9	解析ツール
2 1	サーバ
2 2	クライアント
2 3、2 4	ルータ
I	インターネット

【書類名】 図面

【図 1】



【図 2】



【図 3】

時刻	送信先 IP アドレス	送信元 ポート 番号	あて先 IP アドレス	あて先 ポート 番号	シーケンス 番号	データ 長	確認 応答 番号
15:37:21.026924	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3278262
15:37:21.028260	192.168.1.31	80	192.168.1.48	10637	3320602	1460	3832321198
15:37:21.029493	192.168.1.31	80	192.168.1.48	10637	3322062	1460	3832321198
15:37:21.066928	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.068257	192.168.1.31	80	192.168.1.48	10637	3323522	1460	3832321198
15:37:21.069490	192.168.1.31	80	192.168.1.48	10637	3324982	1460	3832321198
15:37:21.667007	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.668313	192.168.1.31	80	192.168.1.48	10637	3326442	1460	3832321198
15:37:21.697003	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.698301	192.168.1.31	80	192.168.1.48	10637	3279722	1448	3832321198
15:37:21.727007	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.728322	192.168.1.31	80	192.168.1.48	10637	3327902	1460	3832321198
15:37:21.767018	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.768311	192.168.1.31	80	192.168.1.48	10637	3329362	1460	3832321198
15:37:21.797019	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.798324	192.168.1.31	80	192.168.1.48	10637	3330822	1460	3832321198
15:37:21.827024	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.828319	192.168.1.31	80	192.168.1.48	10637	3332282	1460	3832321198
15:37:21.857028	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.858329	192.168.1.31	80	192.168.1.48	10637	3333742	1460	3832321198
15:37:21.887028	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.888322	192.168.1.31	80	192.168.1.48	10637	3335202	1460	3832321198
15:37:21.917040	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.918333	192.168.1.31	80	192.168.1.48	10637	3336662	1460	3832321198
15:37:21.947039	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.948342	192.168.1.31	80	192.168.1.48	10637	3338122	1460	3832321198
15:37:21.987042	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
15:37:21.988337	192.168.1.31	80	192.168.1.48	10637	3339582	1460	3832321198

P1

P2

P3

P4

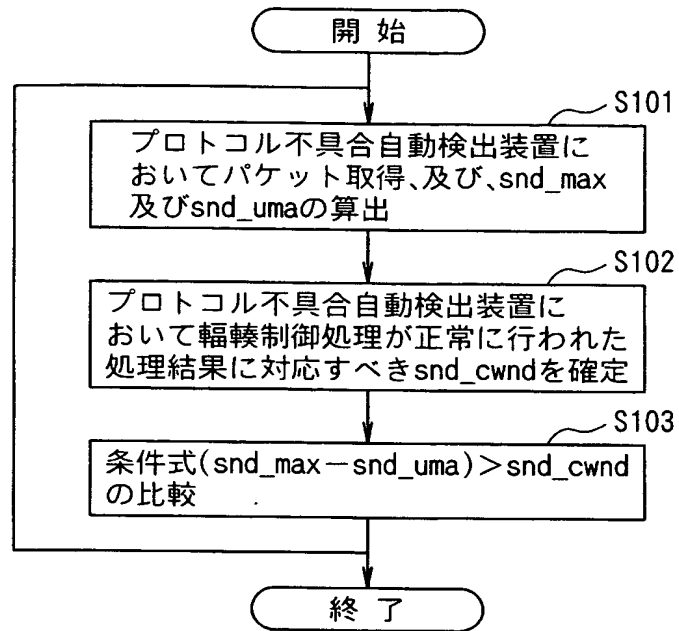
P5

P6

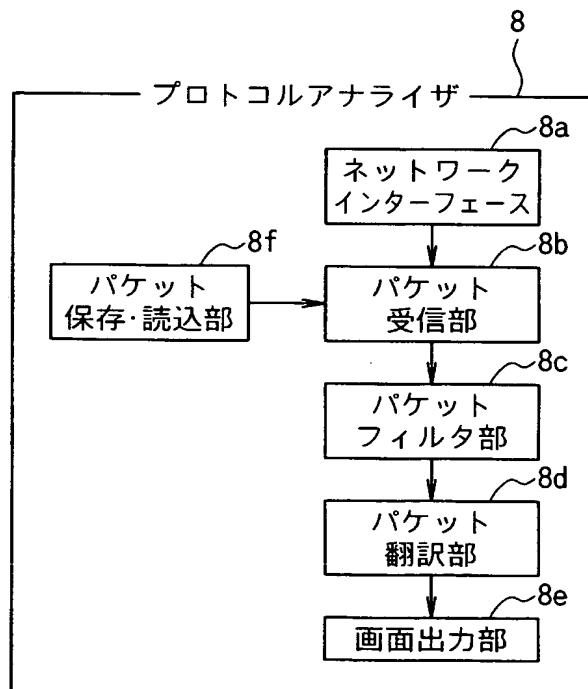
【図 4】

	時刻	snd_max	snd_uma	(snd_max - snd_uma)	snd_cwnd
P1	15:37:21.026924	3319142	3278262	40880	45260
	15:37:21.028260	3320602	3278262	42340	45260
	15:37:21.029493	3322062	3278262	43800	45260
P2	15:37:21.066928	3322062	3279722	42340	46720
	15:37:21.068257	3323522	3279722	43800	46720
	15:37:21.069490	3324982	3279722	45260	46720
P3	15:37:21.667007	3324982	3279722	45260	46720
	15:37:21.668313	3326442	3279722	46720	46720
	15:37:21.697003	3326442	3279722	46720	46720
P4	15:37:21.698301	3326442	3279722	46720	46720
	15:37:21.727007	3326442	3279722	46720	27740
	15:37:21.728322	3327902	3279722	48180	27740
P5	15:37:21.767018	3327902	3279722	48180	29200
	15:37:21.768311	3329362	3279722	49640	29200
	15:37:21.797019	3329362	3279722	49640	30660
P6	15:37:21.798324	3330822	3279722	51100	30660
	15:37:21.827024	3330822	3279722	51100	32120
	15:37:21.828319	3332282	3279722	52560	32120
P7	15:37:21.857028	3332282	3279722	52560	33580
	15:37:21.858329	3333742	3279722	54020	33580
	15:37:21.887028	3333742	3279722	54020	35040
P8	15:37:21.888322	3335202	3279722	55480	35040
	15:37:21.917040	3335202	3279722	55480	36500
	15:37:21.918333	3336662	3279722	56940	36500
P9	15:37:21.947039	3336662	3279722	56940	37960
	15:37:21.948342	3338122	3279722	58400	37960
	15:37:21.987042	3338122	3279722	58400	39420

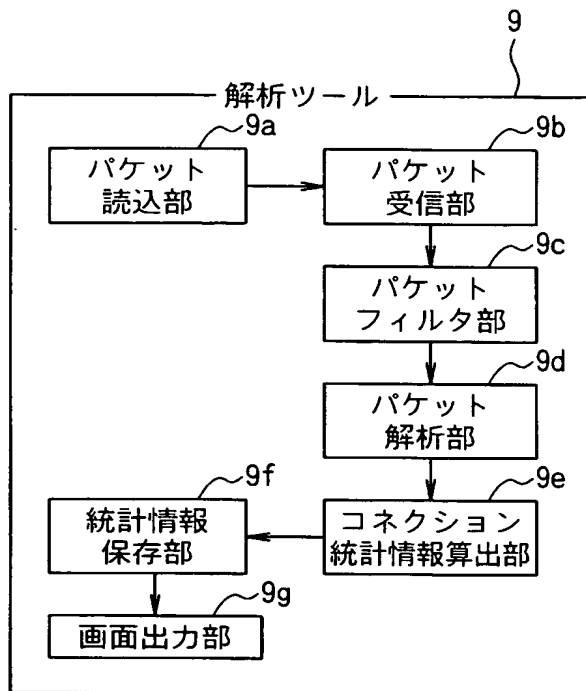
【図 5】



【図 6】



【図 7】



【書類名】 要約書

【要約】

【課題】 本発明の目的は、上述の課題に鑑みてなされたものであり、専門的な知識と煩雑な作業なしにプロトコルの不具合を検出可能なプロトコル不具合自動検出装置を提供する。

【解決手段】 端末間で送受されるパケットをネットワークインターフェース 1 a を介して取得し、該取得されたパケットに基づいて、通信プロトコルに従った送受信制御の結果に対応すべきパケットの送受信状態に関する状態情報を算出する（コネクション情報算出部 1 d）。また、該パケットに基づいて行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する（正常情報算出部 1 f）。当該状態情報と、当該正常情報と、を不具合情報に規定される関係に従って比較（不具合比較判定部 1 h）することにより、プロトコルの不具合を検出することができる。

【選択図】 図 1

特願 2 0 0 2 - 3 6 3 8 9 4

出 願 人 履 歴 情 報

識別番号

[3 9 2 0 2 6 6 9 3]

1. 変更年月日
[変更理由]

2 0 0 0 年 5 月 1 9 日

名称変更

住所変更

住 所
氏 名

東京都千代田区永田町二丁目 1 1 番 1 号
株式会社エヌ・ティ・ティ・ドコモ